Diese Kopfleiste bitte unbedingt ausfüllen!

Familienname, Vorname (bitte durch eine Leerspalte trennen)

Bereich	Berufsnummer	IHK-Nummer	Prüflingsnummer
6 7	1 2 0 1		
Sp. 1-2	Sp. 3-6	Sp. 7-9	Sp. 10-14

Termin: Mittwoch, 24. November 2021

IHK

Abschlussprüfung Winter 2021/22

1201

2 Entwicklung und Umsetzung von Algorithmen

Fachinformatiker
Fachinformatikerin
Anwendungsentwicklung (AO 2020)

Teil 2 der Abschlussprüfung

4 Aufgaben
mit Belegsatz
90 Minuten Prüfungszeit
100 Punkte

Hinweis:
Bei der Bearbeitung der Aufgaben ist von einem gewöhnlichen Geschäftsbetrieb auszugehen, der **nicht** durch die COVID-19-Pandemie beeinflusst bzw. durch entsprechende behördliche Verfügungen eingeschränkt ist.

Bearbeitungshinweise

1. Bevor Sie mit der Bearbeitung der Aufgaben beginnen, überprüfen Sie bitte die **Vollständigkeit** dieses Aufgabensatzes. Die Anzahl der zu bearbeitenden Aufgaben ist auf dem Deckblatt links angegeben. Wenden Sie sich bei Unstimmigkeiten sofort an die Aufsicht, weil Reklamationen am Ende der Prüfung nicht anerkannt werden können.
2. Füllen Sie zuerst die **Kopfzeile** aus. Tragen Sie Ihren Familiennamen, Ihren Vornamen und Ihre Prüflings-Nr. in die oben stehenden Felder ein.
3. Lesen Sie bitte den **Text** der Aufgaben ganz durch, bevor Sie mit der Bearbeitung beginnen.
4. Halten Sie sich bei der Bearbeitung der Aufgaben genau an die **Vorgaben der Aufgabenstellung** zum Umfang der Lösung. Wenn z. B. vier Angaben gefordert werden und Sie sechs Angaben anführen, werden nur die ersten vier Angaben bewertet.
5. Tragen Sie die frei zu formulierenden **Antworten dieser offenen Aufgaben** in die dafür lt. Aufgabenstellung vorgesehenen Bereiche (Lösungszeilen, Formulare, Tabellen u. a.) des Arbeitsbogens ein.
6. Sofern nicht ausdrücklich ein Brief oder eine Formulierung in ganzen Sätzen gefordert werden, ist eine **stichwortartige Beantwortung** zulässig.
7. Schreiben Sie deutlich und gut lesbar. Ein nicht eindeutig zuzuordnendes oder **unleserliches Ergebnis** wird als **falsch** gewertet.
8. Zur Lösung der Rechenaufgaben darf ein nicht programmierter, netzunabhängiger **Taschenrechner** ohne Kommunikationsmöglichkeit mit Dritten verwendet werden.
9. Wenn Sie ein **gerundetes Ergebnis** eintragen und damit weiterrechnen müssen, rechnen Sie (auch im Taschenrechner) nur mit diesem gerundeten Ergebnis weiter.
10. Für **Nebenrechnungen/Hilfsaufzeichnungen** können Sie das im Aufgabensatz enthaltene Konzeptpapier verwenden. Dieses muss vor der Bearbeitung der Aufgaben herausgetrennt werden. Bewertet werden jedoch nur Ihre Eintragungen im Aufgabensatz.

Wird vom Korrektor ausgefüllt!

Bewertung

Für die Bewertung gilt die Vorgabe der Punkte in den Lösungshinweisen.

1. Aufg.		Punkte	2. Aufg.		Punkte	3. Aufg.		Punkte	4. Aufg.		Punkte
	15 16			17 18			19 20			21 22	

Prüfungszeit [ ] (23)

Die entsprechende Ziffer (1, 2 oder 3) finden Sie in der Abfrage nach der Prüfungszeit im Anschluss an die letzte Aufgabe.

Gesamtpunktzahl [ ] [ ] [ ] (24 25 26)

Prüfungsort, Datum

Unterschrift

Gemeinsame Prüfungsaufgaben der Industrie- und Handelskammern. Dieser Aufgabensatz wurde von einem überregionalen Ausschuss, der entsprechend § 40 Berufsbildungsgesetz zusammengesetzt ist, beschlossen.
Die Vervielfältigung, Verbreitung und öffentliche Wiedergabe der Prüfungsaufgaben und Lösungen ist nicht gestattet. Zuwiderhandlungen werden zivil- und strafrechtlich (§§ 97 ff., 106 ff. UrhG) verfolgt. – © ZPA Nord-West 2021 – Alle Rechte vorbehalten!

Die Aufgaben 1 bis 4 beziehen sich auf die folgende Ausgangssituation:

Sie arbeiten im Projekt zur Entwicklung einer Software zur Terminvergabe in Arztpraxen mit. Dabei gehört es zu Ihren Aufgaben, bei der Planung Sicherheitsaspekte einzubeziehen sowie Teststrategien herauszuarbeiten.

Korrekturrand

1. Aufgabe (25 Punkte)

Sie erhalten den Auftrag die „Online-Terminvergabe" grundlegend zu planen. Dazu wurden zwei unterschiedliche Vorgehensweisen grob vorgeplant. Diese sind in der Tabelle „Ablauf" mit Variante 1 und Variante 2 dargestellt.

a) Stellen Sie den Ablauf der Variante 1 mit einem UML-Aktivitätsdiagramm dar. 15 Punkte

Ablauf:

Patient ruft die Terminreservierung auf, wonach er eine Anzeige der Ärzte und Terminarten erhält. Jetzt kann er den Arzt bzw. die Terminart auswählen. Nach der Auswahl sucht die Software den 1. freien Termin. ...	
Variante 1:	Variante 2:
... Das Programm zeigt den Monat, in dem der 1. freie Termin liegt, mit allen freien Terminen im angezeigten Zeitraum an. Der Anwender wählt einen freien Termin. Das System versucht nun, den Termin zu blockieren. Wenn der Termin noch frei ist, kann er final reserviert und parallel dazu der Anwender informiert werden. Danach ist die Aktivität beendet. Ist der Termin inzwischen nicht mehr verfügbar, wird eine entsprechende Meldung ausgeben und erneut der 1. freie Termin gesucht, um dem Anwender die Auswahl anzuzeigen.	... Die Software ermittelt den Monat des 1. freien Termins. Es werden alle Termine des ermittelten Monats blockiert. Der Monat mit allen freien Terminen wird zur Auswahl angezeigt. Der Anwender kann einen freien Termin wählen. Danach wird der Anwender bzgl. der erfolgreichen Reservierung informiert und parallel der Termin final reserviert sowie alle anderen blockierten Termine freigegeben. Danach ist die Aktivität beendet.

UML-Aktivitätsdiagramm:

Anwender	Terminvergabesystem

b) Wählen Sie für die zu erstellende Terminvergabe eine Variante aus und begründen Sie Ihre Wahl in Hinblick auf die Praxistauglichkeit. 4 Punkte

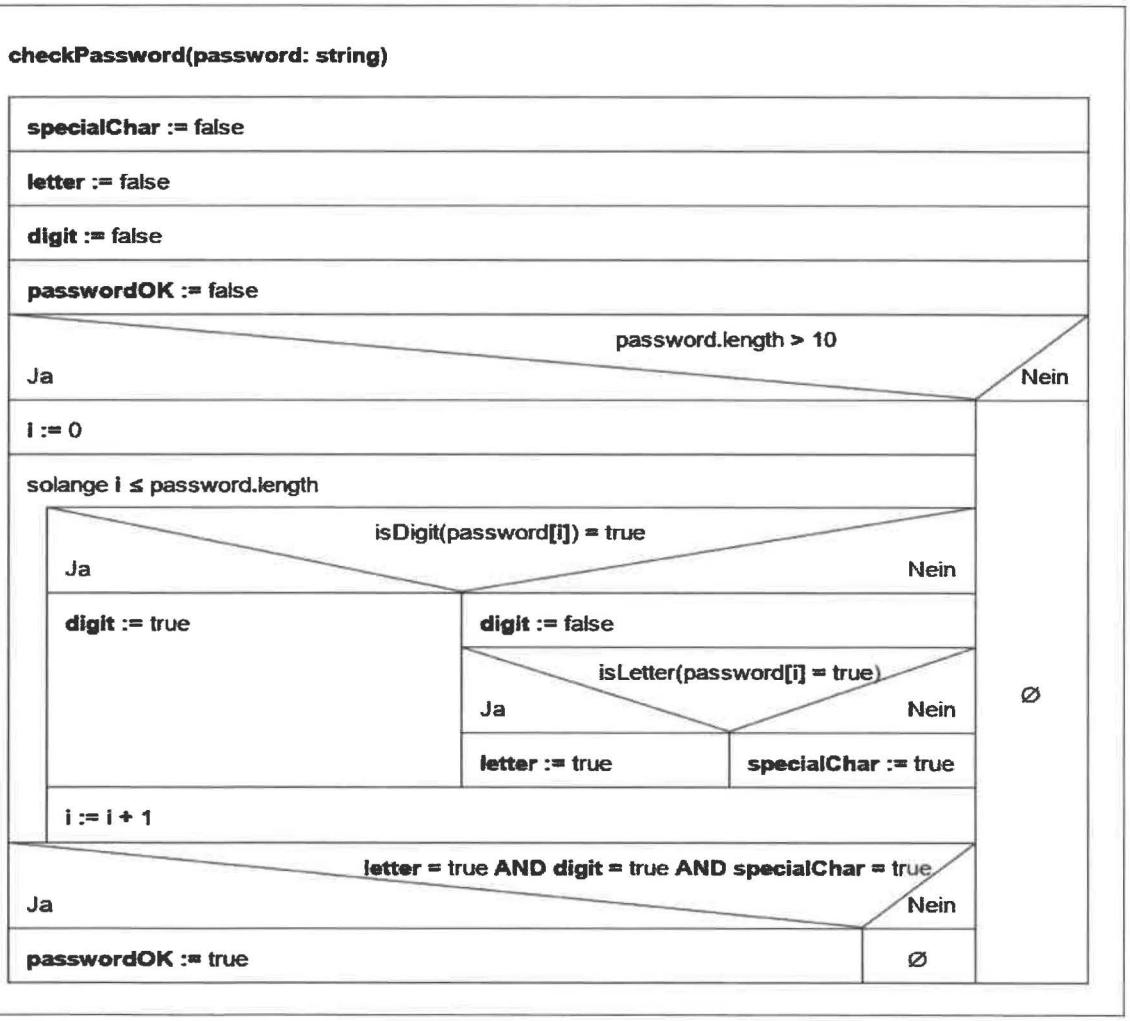
Korrekturrand

c) Die Online-Terminvergabe soll nur über einen Login möglich sein. Dazu müssen die Benutzer einen Account erstellen und ein Passwort vergeben, welches folgende Kriterien erfüllen muss:

- Es muss mindestens zehn Zeichen lang sein.
- Es muss mindestens einen Buchstaben (letter), eine Ziffer (digit) und ein Sonderzeichen (specialChar) enthalten.

Es liegt ein fehlerhaftes Struktogramm zur Überprüfung des Passwortes vor. Der Rückgabewert soll wahr oder falsch sein.

Finden Sie drei Fehler und geben Sie eine Korrektur an. 6 Punkte



2. Aufgabe (25 Punkte)

Korrekturrand

Für eine Gemeinschaftspraxis, der mehrere Ärzte angehören, soll für die Sprechstundenhilfen eine softwaregestützte Terminplanung realisiert werden.

Benötigt wird dazu eine Funktion, die unter Vorgabe eines Datums (Tag und Monat), einer Ärzte-ID und eines Zeitslots ermittelt, ob unter diesen Vorgaben ein Termin möglich ist.

Für jeden Arzt liegt ein Terminplan in Form eines zweidimensionalen Arrays vor. Jede Zeile dieses Arrays speichert für einen Tag die Belegung von acht möglichen Zeitslots für diesen Arzt.

Beispiel:

Monat	Tag	Slot 0	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
...									
9	30	1	1	0	0	1	1	1	1
10	1	0	0	1	1	0	0	1	1
10	2	0	0	0	1	1	1	1	1
...									

(1 – Arzt in diesem Zeitslot belegt, 0 – Arzt in diesem Slot nicht belegt)

Für den Zugriff auf das zweidimensionale Array eines Arztes liegt folgende Funktion bereits vor:

getTimeTable (id : Integer) : 2-dim-Integer-Array	Liefert für eine übergebene Ärzte-ID den zugehörigen Terminplan als zweidimensionales Array (siehe Beispiel)

Es soll folgende Funktion erstellt werden:

getAppointment(month: Integer, day: Integer, id : Integer, slot: Integer) : boolean

- Falls kein bestimmter Zeitslot ausgewählt werden muss, hat der Parameter „slot" den Wert -1.
- Die Funktion *getAppointment* soll den Wert ***true*** zurückgeben, falls ein Termin gefunden werden kann, andernfalls den Wert ***false***.

Erstellen Sie auf der gegenüberliegenden Seite diese Funktion *getAppointment* in Pseudocode.

getAppointment(month: Integer, day: Integer, id : Integer, slot: Integer) : boolean

Korrekturrand

3. Aufgabe (25 Punkte)

Korrekturrand

Ein Kollege von Ihnen hatte begonnen, das relationale Modell der bestehenden Datenbank zu erweitern. Krankheitsbedingt fällt er leider nun einige Wochen aus und Sie übernehmen seine Aufgaben. Zunächst sollen Sie das relationale Datenmodell fertigstellen. Dazu stehen Ihnen die nachfolgenden Unterlagen zur Verfügung. In diesen sind exemplarische Daten für Untersuchungstermine und Patientendiagnosen enthalten. Diese Daten sollen in der Datenbank verwaltet werden.

Untersuchungstermine

UTNr	Datum/Uhrzeit	Patient	Arzt	Raum
1	20.04.2021 – 10.00 Uhr	Sommer, Hans Hohlweg 17, 55555 Bergtal	Baum, Walter	Röntgenraum 2
2	20.04.2021 – 11.00 Uhr	Reins, Fraucke Winzergasse 5, 55555 Bergtal	Baum, Walter	Röntgenraum 1
3	21.04.2021 – 10.00 Uhr	Fuchs, Katrin Bachweg 12, 55554 Zweigtal	Winter, Simone	MRT

Patientendiagnose

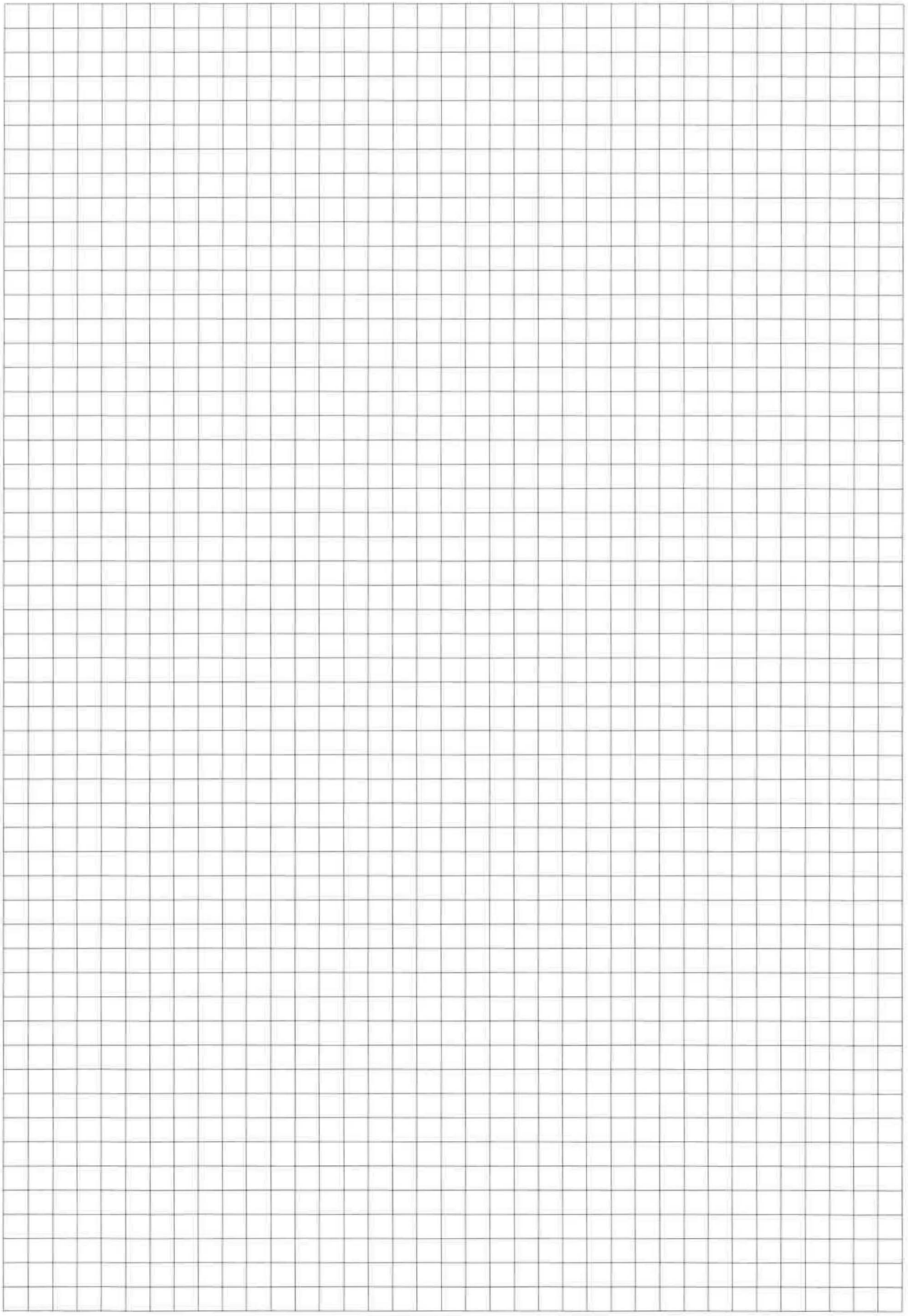
PDNr	Datum	Patient	Arzt	ICD_10_GM_Code	Bermerkungen
1	23.04.2021	Sommer, Hans	Baum, Walter	M71	Wiedervorstellung in 3 Monaten
2	07.06.2021	Reins, Fraucke	Frühling, Klaus	K93	
3	22.06.2021	Fuchs, Katrin	Winter, Simone	D50	Überweisung an einen Spezialisten

Hinweis: Die Untersuchungstermine und die Patientendiagnose sind voneinander unabhängig. Die Diagnose kann auch ein anderer Arzt als der untersuchende Arzt erstellen.

Erweitern Sie das relationale Datenmodell auf Seite 7 um die notwendigen Tabellen mit den entsprechenden Attributen in der 3. NF. Zeichnen Sie die Beziehungen zwischen den Tabellen ein und bestimmen Sie die Kardinalitäten. Kennzeichnen Sie in allen Tabellen die Primär- und Fremdschlüssel mit PK bzw. FK.

Relationales Modell:

Diagnose
DiagnoseID (PK) ICD_10_GM_Code Kurzbeschreibung

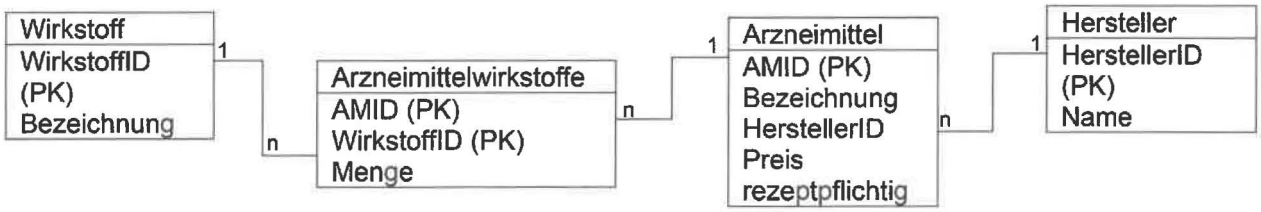


Korrekturrand

4. Aufgabe (25 Punkte)

Korrekturrand

Die Arzneimittel mit deren Wirkstoffen sind in folgender Datenbank erfasst.



Hinweis: Das Attribut „rezeptpflichtig" hat den Datentyp Boolean. Die Überschriften müssen nicht per Alias angepasst werden.

Erstellen Sie für die nachfolgenden Aufgabenstellungen die entsprechenden SQL-Anweisungen.

a) Legen Sie die Tabelle „Hersteller" an. 3 Punkte

b) Zeigen Sie an, ob das Arzneimittel mit der Bezeichnung „abc" rezeptpflichtig ist. 3 Punkte

c) Erhöhen Sie den Preis aller Arzneimittel vom Hersteller „CCC" um fünf Prozent. 5 Punkte

d) Listen Sie alle Hersteller auf, von denen mehr als 50 nichtrezeptpflichtige Arzneimittel in der Datenbank erfasst sind. Die Ausgabe soll folgende Daten auflisten. 6 Punkte

Korrekturrand

Beispiel:

Hersteller	Anzahl
ABC	234
DDDD	96
XXX	51

e) Listen Sie die Bezeichnung, die Menge, den Preis und den Hersteller aller Arzneimittel auf, welche den Wirkstoff „Y34" in der Menge von 100 oder 200 Einheiten enthalten. Das Ergebnis soll absteigend nach der Menge und dann aufsteigend nach dem Preis sortiert sein. 8 Punkte

Beispiel:

Bezeichnung	Menge	Preis	Hersteller
AY34-200	200	13,67	AAA
BB Y34	200	15,88	BBB
AY34-100	100	8,55	AAA
CC XX 100	100	9,99	CCC

Fortsetzung 4. Aufgabe →

Fortsetzung 2. Aufgabe

Korrekturrand

PRÜFUNGSZEIT – NICHT BESTANDTEIL DER PRÜFUNG!

Wie beurteilen Sie nach der Bearbeitung der Aufgaben die zur Verfügung stehende Prüfungszeit?

[1] Sie hätte kürzer sein können.
[2] Sie war angemessen.
[3] Sie hätte länger sein müssen.

Abschlussprüfung Winter 2021/22
Lösungshinweise

Fachinformatiker/Fachinformatikerin
Anwendungsentwicklung (AO 2020)
1201

2 Entwicklung und Umsetzung von Algorithmen

Teil 2 der Abschlussprüfung

Allgemeine Korrekturhinweise

Die Lösungs- und Bewertungshinweise zu den einzelnen Handlungsschritten sind als Korrekturhilfen zu verstehen und erheben nicht in jedem Fall Anspruch auf Vollständigkeit und Ausschließlichkeit. Neben hier beispielhaft angeführten Lösungsmöglichkeiten sind auch andere sach- und fachgerechte Lösungsalternativen bzw. Darstellungsformen mit der vorgesehenen Punktzahl zu bewerten. Der Bewertungsspielraum des Korrektors (z. B. hinsichtlich der Berücksichtigung regionaler oder branchenspezifischer Gegebenheiten) bleibt unberührt.

Zu beachten ist die unterschiedliche Dimension der Aufgabenstellung (nennen – erklären – beschreiben – erläutern usw.).

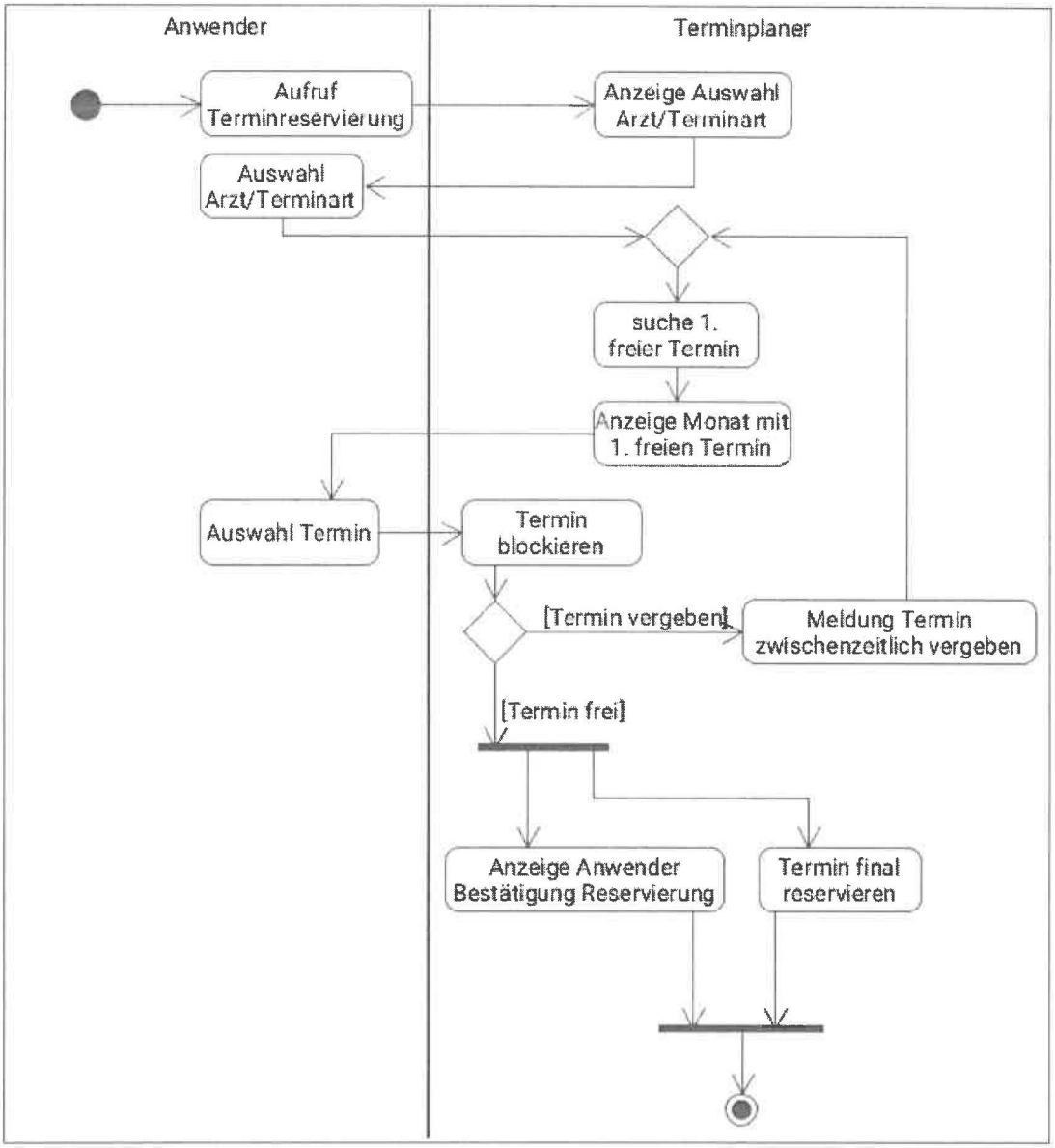
Für die Bewertung gilt folgender Punkte-Noten-Schlüssel:

Note 1	=	100 – 92 Punkte	Note 2	= unter	92 – 81 Punkte
Note 3	= unter	81 – 67 Punkte	Note 4	= unter	67 – 50 Punkte
Note 5	= unter	50 – 30 Punkte	Note 6	= unter	30 – 0 Punkte

Die Vervielfältigung, Verbreitung und öffentliche Wiedergabe der Prüfungsaufgaben und Lösungen ist nicht gestattet. Zuwiderhandlungen werden zivil- und strafrechtlich (§§ 97 ff., 106 ff. UrhG) verfolgt. – © ZPA Nord-West 2021 – Alle Rechte vorbehalten!

1. Aufgabe (25 Punkte)

a) 15 Punkte



Pro Aktivität 0,5 Punkte: 5 Punkte
Parallelisierung: 2 Punkte
Verzweigung mit Bedingungen: 3 Punkte
Anordnung in Partitionen, Start und Ende, Reihenfolge korrekt: 5 Punkte
Alternative Lösungen sind möglich.

b) 4 Punkte
Auswahl: Variante 1
Begründung: Es ist davon auszugehen, dass in der Praxis häufig gleichzeitige Zugriffe erfolgen. In der Variante 2 wären die Termine eines Monats unnötig lange blockiert.

Variante 2 bei entsprechender Begründung ist auch möglich.

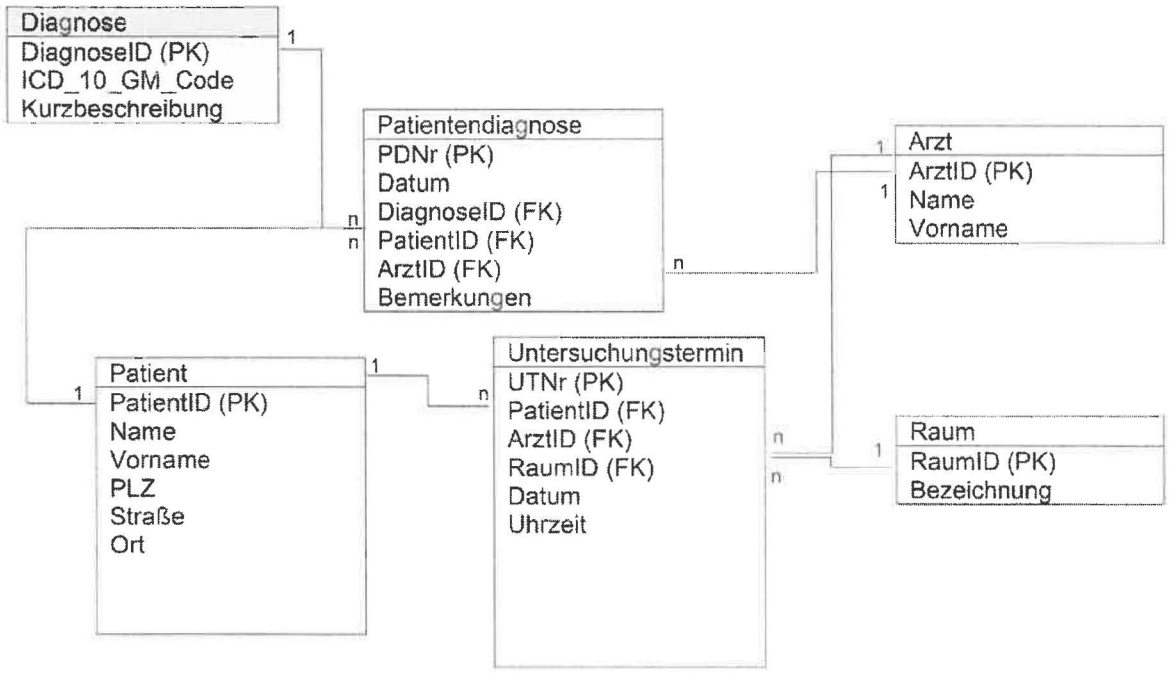
c) 6 Punkte
1. Fehler: Die Verzweigungsbedingung password.length > 10 muss password.length >= 10 lauten.
2. Fehler: Die Schleifenbedingung i <= password.length muss i < password.length lauten.
3. Fehler: Die Anweisung digit := false im Nein-Zweig der Verzweigung ist zu streichen.
4. Fehler: Es fehlt die Angabe des Return-Wertes.

2. Aufgabe (25 Punkte)

```
getAppointment(month: Integer, day: Integer, id: Integer,
               time_slot: Integer) : boolean

      boolean gefunden = false
      int[][] timeTableDoctor = getTimeTable(id)

      für i = 0, i < timeTableDoctor.length

            wenn (month == timeTableDoctor[i][0] &&
                  day == timeTableDoctor[i][1])

                  wenn (time_slot < 0)
                        für j = 0; j < 8
                              wenn (timeTableDoctor[i][2 + j] == 0)
                                    gefunden = true
                              ende wenn
                        ende für
                  sonst
                        wenn (timeTableDoctor[i][2 + time_slot] == 0)
                              gefunden = true
                        ende wenn
                  ende wenn
            ende wenn
      ende für

      Rückgabe gefunden;

ende getAppointment
```

3. Aufgabe (25 Punkte)

Punktevergabe:
Für die Tabellen (Raum, Arzt) mit allen Attributen je 1 Punkt = 2 Punkte
Für die Tabellen (Patient, Untersuchungstermin, Patientendiagnose) mit allen Attributen je 2 Punkte = 6 Punkte
Pro Primärschlüssel 1 Punkt = 5 Punkte
Pro Fremdschlüssel 1 Punkt = 6 Punkte
Pro Beziehung/Kardinalität 1 Punkt = 6 Punkte



Auch andere Lösungen sind möglich.

4. Aufgabe (25 Punkte)

a) 3 Punkte

```
CREATE TABLE Hersteller (HerstellerID INT, Bezeichnung VARCHAR(30),
                         PRIMARY KEY (HerstellerID));
```

b) 3 Punkte

```
SELECT rezeptpflichtig FROM Arzneimittel WHERE Bezeichnung = "abc";
```

c) 5 Punkte

```
UPDATE Arzneimittel SET Preis = Preis * 1.05
WHERE HerstellerID = (SELECT HerstellerID FROM Hersteller WHERE Bezeichnung = "CCC")
```

d) 6 Punkte

```
SELECT H.Bezeichnung, COUNT(A.ArzneimittelID)
FROM Arzneimittel AS A, Hersteller AS H
WHERE A.HerstellerID = H.HerstellerID AND rezeptpflichtig = False
GROUP BY H.Bezeichnung HAVING COUNT(A.ArzneimittelID) > 50
```

oder

```
SELECT H.Bezeichnung, COUNT(A.ArzneimittelID) FROM Arzneimittel AS A INNER JOIN Hersteller AS H ON A.HerstellerID = H.HerstellerID
WHERE rezeptpflichtig = False
GROUP BY H.Bezeichnung HAVING COUNT(A.ArzneimittelID) > 50
```

e) 8 Punkte

```
SELECT A.Bezeichnung, AW.Menge, A.Preis, H.Bezeichnung
FROM Wirkstoff AS W, Arzneimittelwirkstoffe AS AW, Arzneimittel AS A, Hersteller AS H
WHERE W.WirkstoffID = AW.WirkstoffID AND
      AW.AMID = A.AMID AND
      A.HerstellerID = H.HerstellerID AND
      W.Bezeichnung = "Y34" AND (AW.Menge = 100 OR AW.Menge = 200)
ORDER BY AW.Menge DESC, A.Preis ASC
```

Auch andere Lösungen sind möglich.